**Task 1. Create a New Book Record** -- "978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes', 'Harper Lee', 'J.B. Lippincott & Co.')"

**Task 2: Update an Existing Member's Address**

**Task 3: Delete a Record from the Issued Status Table** -- Objective: Delete the record with issued\_id = 'IS121' from the issued\_status table.

**Task 4: Retrieve All Books Issued by a Specific Employee** -- Objective: Select all books issued by the employee with emp\_id = 'E101'.

**Task 5: List Members Who Have Issued More Than One Book** -- Objective: Use GROUP BY to find members who have issued more than one book.

**Task 6: Create Summary Tables**: Used CTAS to generate new tables based on query results - each book and total book\_issued\_cnt\*\*

**Task 7: Retrieve All Books in a Specific Category**:

**Task 8: Find Total Rental Income by Category**:

Task 9: **List Members Who Registered in the Last 180 Days**:

**Task 10: List Employees with Their Branch Manager's Name and their branch details**:

Task 11. **Create a Table of Books with Rental Price Above a Certain Threshold**:

Task 12: **Retrieve the List of Books Not Yet Returned**

**Advanced SQL Operations**

**Task 13: Identify Members with Overdue Books**  
Write a query to identify members who have overdue books (assume a 30-day return period). Display the member's\_id, member's name, book title, issue date, and days overdue.

**Task 14: Update Book Status on Return**  
Write a query to update the status of books in the books table to "Yes" when they are returned (based on entries in the return\_status table).

**Task 15: Branch Performance Report**  
Create a query that generates a performance report for each branch, showing the number of books issued, the number of books returned, and the total revenue generated from book rentals.

**Task 16: CTAS: Create a Table of Active Members**  
Use the CREATE TABLE AS (CTAS) statement to create a new table active\_members containing members who have issued at least one book in the last 2 months.

**Task 17: Find Employees with the Most Book Issues Processed**  
Write a query to find the top 3 employees who have processed the most book issues. Display the employee name, number of books processed, and their branch.

**Task 18: Identify Members Issuing High-Risk Books**  
Write a query to identify members who have issued books more than twice with the status "damaged" in the books table. Display the member name, book title, and the number of times they've issued damaged books.

**Task 19: Stored Procedure** Objective: Create a stored procedure to manage the status of books in a library system. Description: Write a stored procedure that updates the status of a book in the library based on its issuance. The procedure should function as follows: The stored procedure should take the book\_id as an input parameter. The procedure should first check if the book is available (status = 'yes'). If the book is available, it should be issued, and the status in the books table should be updated to 'no'. If the book is not available (status = 'no'), the procedure should return an error message indicating that the book is currently not available.

**Task 20: Create Table As Select (CTAS)** Objective: Create a CTAS (Create Table As Select) query to identify overdue books and calculate fines.

Description: Write a CTAS query to create a new table that lists each member and the books they have issued but not returned within 30 days. The table should include: The number of overdue books. The total fines, with each day's fine calculated at $0.50. The number of books issued by each member. The resulting table should show: Member ID Number of overdue books Total fines